Lab Exercise 1: Simple Testbench for Circuit 1
(Combinational logic)

Develop a testbench in VHDL to verify the operation of an ALU (Arithmetic-Logic Unit) specified using Fig. 1 and Tables 1 and 2 below.

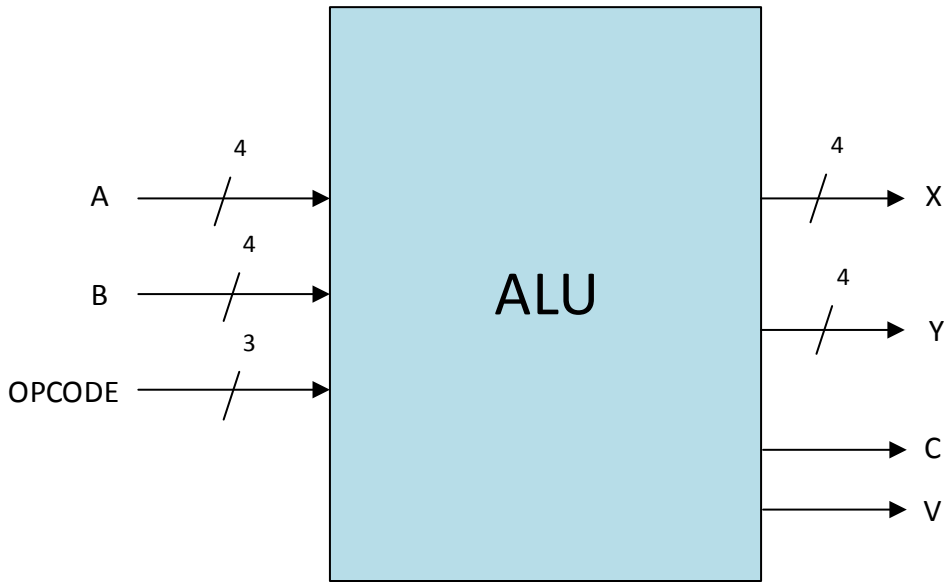

Fig. 1 Interface of ALU

Table 1. Meaning and size of inputs and outputs

Name	Mode	Width	Meaning
A	INPUT	4	Input A
B	INPUT	4	Input B
OPCODE	INPUT	3	Operation Code
X	OUTPUT	4	Output or Least Significant Nibble (4-bits) of Output
Y	OUTPUT	4	Most Significant Nibble of Output [used only during multiplication]
C	OUTPUT	1	Carry Out
V	OUTPUT	1	Overflow

Table 2. Meaning of Opcodes

OPCODE	OPERATION	FORMULA	Y	V	C
000	NOR	X = A NOR B	0	0	0
001	NAND	X = A NAND B	0	0	0
010	XOR	X = A XOR B	0	0	0
011	Unsigned Addition	(C:X) = A + B	0	0	↕
100	Signed Addition	X = A + B	0	↕	0
101	Signed Subtraction	X = A - B	0	↕	0
110	Unsigned Multiplication	(Y:X) = A * B	↕	0	0
111	Signed Multiplication	(Y:X) = A * B	↕	0	0

The notation used in Table 2 is as follows:
(Y:X) denotes concatenation of Y and X, with the most significant part stored in Y, and the least significant part stored in X.
↕ means that a given output is affected.
0 means that a given output is set to 0.

For each Opcode, your testbench should provide two sets of inputs, as specified in Table 3.

Table 3. Required values of inputs in the hexadecimal notation.

OPCODE	OPERATION	FORMULA	A	B
000	NOR	X = A NOR B	0	C
000	NOR	X = A NOR B	1	4
001	NAND	X = A NAND B	2	3
001	NAND	X = A NAND B	3	A
010	XOR	X = A XOR B	4	7
010	XOR	X = A XOR B	5	9
011	Unsigned Addition	(C:X) = A + B	6	9
011	Unsigned Addition	(C:X) = A + B	7	A
100	Signed Addition	X = A + B	8	7
100	Signed Addition	X = A + B	9	8
101	Signed Subtraction	X = A - B	A	B
101	Signed Subtraction	X = A - B	B	D
110	Unsigned Multiplication	(Y:X) = A * B	C	4
110	Unsigned Multiplication	(Y:X) = A * B	D	2
111	Signed Multiplication	(Y:X) = A * B	E	E
111	Signed Multiplication	(Y:X) = A * B	F	F

Each set of inputs should be provided for a period of time defined by the constant `period` with the value of 10 ns.

Write and debug your testbench.

Then, use it to verify the correctness of two post-synthesis simulation models (black boxes):

ALU_synthesis_1.vhd and ALU_synthesis_2.vhd.

One of these models contains errors. Please find out which model is correct and which contains errors. Then, locate at least 3 errors in the faulty model.

Tasks:

1. Supplement missing values of expected outputs in Table 4.

2. Develop VHDL code of the testbench.

3. Verify the correctness of input waveforms for the inputs OPCODE, A, and B generated by your testbench.

4. Add an instantiation of ALU, compile your testbench together with

    a. ALU_synthesis_1.vhd

    b. ALU_synthesis_2.vhd

   and verify the operation of each model separately.

5. Determine a list of at least 3 errors in the faulty model.

Table 4. Expected values of outputs.

OPCODE	OPERATION	FORMULA	A	B	X	Y	C	V
000	NOR	X = A NOR B	0	C		0	0	0
000	NOR	X = A NOR B	1	4		0	0	0
001	NAND	X = A NAND B	2	3		0	0	0
001	NAND	X = A NAND B	3	A		0	0	0
010	XOR	X = A XOR B	4	7		0	0	0
010	XOR	X = A XOR B	5	9		0	0	0
011	Unsigned Addition	(C:X) = A + B	6	9		0		0
011	Unsigned Addition	(C:X) = A + B	7	A		0		0
100	Signed Addition	X = A + B	8	7		0	0	
100	Signed Addition	X = A + B	9	8		0	0	
101	Signed Subtraction	X = A - B	A	B		0	0	
101	Signed Subtraction	X = A - B	B	D		0	0	
110	Unsigned Multiplication	(Y:X) = A * B	C	4			0	0
110	Unsigned Multiplication	(Y:X) = A * B	D	2			0	0
111	Signed Multiplication	(Y:X) = A * B	E	E			0	0
111	Signed Multiplication	(Y:X) = A * B	F	F			0	0